ORIGINAL RESEARCH

Diagnosis of COVID-19 CT Scans Using Convolutional Neural Networks

Victor Chang[1] · Siddharth Mcwann[1] · Karl Hall[2] · Qianwen Ariel Xu[1] · Meghana Ashok Ganatra[1]

Received: 2 December 2023 / Accepted: 5 April 2024

Abstract
Machine learning technology, particularly neural networks, provides useful tools for diagnosing diseases. This study focuses on how convolutional neural networks can be implemented to diagnose COVID-19 through the processing of x-ray images. This study demonstrates how the convolutional neural networks DenseNet201, ResNet152, VGG16, and InceptionV3 can aid healthcare providers in the diagnosis of COVID-19. The models returned accuracies of 98.73%, 97.23%, 91.25% and 98.38% respectively. The results from these experiments are compared to previous studies by evaluating F1-score, accuracy, precision and recall. Additionally, the important problems of hyperparameter tuning and data imbalance are explored and addressed. Areas for future research in this area are also suggested.

Keywords COVID-19 · Healthcare systems · Machine learning · Artificial intelligence · Diagnosis · Convolutional neural networks

Introduction

"The current COVID-19 pandemic is caused by a coronavirus named SARS-CoV-2." [9, 21]. Coronaviruses are a large cohort of viruses that spread as viral infections in the respiratory system in humans. The viruses cause the common cold but can also lead to diseases with severe and debilitating symptoms that have high mortality rates. Severe Acute Respiratory Syndrome (SARS), detected in 2003 and Middle East respiratory syndrome (MERS), detected in 2012, are examples of such diseases.

Coronavirus (COVID-19) sickness is a highly contagious viral ailment that first appeared in December 2019. It is caused by coronavirus-2, which causes severe acute respiratory syndrome (SARS-CoV-2). Over 700 million confirmed cases of COVID-19 have been reported to WHO worldwide since the outbreak of the pandemic (World Health Organization 2024). This is a massive number of cases in a span of 4 years and demonstrates that the virus is highly contagious. The health systems of most nations are not adequately equipped to diagnose and treat a virus that is spreading in the population at such an unprecedented rate. Moreover, while adapting to new human hosts, SARS-CoV-2 is prone to genetic evolution with the development of mutations over time, resulting in mutant variants that may have different characteristics from its ancestral strains. Several variants of this virus have emerged since its breakout. However, WHO considers only five variants to have a relentless impact on the population worldwide—Alpha, Beta, Gamma, Delta, and Omicron. Out of these five, Delta and Omicron are relatively new and considered more fatal and contagious, and efforts are still underway to deal with these variants effectively. Notwithstanding, COVID-19 should be considered as disease that needs immediate attention.

Many methods have been developed to diagnose and screen for COVID-19. The paper 'COVID-19 Diagnostic Methods and Detection Techniques: A Review' by Rong et al. [12] discusses many diagnostic testing methods—PCR-based testing methods, isothermal nucleic acid amplification-based methods, CRISPR-based tests, gene sequencing, antibody testing, antigen testing, computed tomography (CT), and biosensor-based detection. They put forward the fact that RT-PCR-based testing methods have been especially effective in diagnosing the disease. However, the supply of

This article is part of the topical collection "Emerging Technologies and Services for post-COViD-19 Era" guest editedby Victor Chang, Gary Wills, Flavia Delicato and Mitra Arami.

✉ Victor Chang
v.chang1@aston.ac.uk; victorchang.research@gmail.com

[1] Department of Operations and Information Management, Aston Business School, Aston University, Birmingham, UK

[2] School of Computing, Engineering and Digital Technologies, Teesside University, Middlesbrough, UK

Published online: 07 June 2024

RT-PCR kits is not sufficient to commensurate with the rising cases of COVID-19. Furthermore, the effectiveness of these kits depends on the presence of SARS-CoV-2 in the collected samples [12]. They suggest a combination of CT scan diagnosis along with RT-PCR to achieve more accurate results [1]. In the case of CT scan image diagnosis, Rong et al. [12] observe that CT scans require operation by professional doctors and that artificial intelligence and machine learning techniques (Fig. 1) should be used to overcome and this drawback by supplementing existing methods [9, 20].

Need for Research

In the event of this pandemic, healthcare workers cannot afford to spend their time on the diagnosis of COVID-19 for a large number of patients. Hence, it is of primary importance to develop a robust method to diagnose the disease in a short span of time. Researchers, scientists, and mathematicians have been working on building automation in health systems to achieve better patient health outcomes. The rapid evolution in machine learning algorithms, artificial intelligence, cloud computing, publicly available annotated datasets and persistent efforts by researchers worldwide have led to the development of processes that can help clinicians achieve effective health outcomes. For example, AI algorithms can be used in the field of bowel polyp identification to help with decisions over segregating normal and abnormal polyps, preventing inappropriate referrals which absorb a huge amount of time, or managing treatment costs for patient and staff well-being, or monitoring patients' conditions in their homes and provide timely interventions when their vital signs warrant them.

Aims & Objectives

The focus of this paper is to look at the future of healthcare with improved patient health outcomes, where people have access to essential health services, and how AI and ML can play a pivotal role in achieving this goal. This research is centered on evaluating the performance of different convolutional neural network (CNN) algorithms in detecting the presence of Omicron and Delta variants of the COVID-19 virus. This paper aims to contribute to the preliminary work that can be used as a reference by future researchers and healthcare workers to judge the effectiveness and suitability of ML methods for the diagnosis of COVID-19.

Additionally, this research also aims to understand the evolution of the healthcare system around the world and the short-term and long-term demands that health systems are required to meet. This can include changes in business models and the adoption of new and evolving practices by healthcare providers.

Paper Structure

The paper is structured as follows:

Sect. "Literature Review" provides a comprehensive literature review on CNN models, CNN data modeling techniques and applications of CNN models in the healthcare domain. Sect. "Methodology" provides a theoretical background of the DenseNet201, ResNet152, VGG16, and InceptionV3 models explored in the study. Other details about the methodology, such as data collection, data preprocessing, hyperparameter tuning, and model training, are also highlighted in this section. Sect. "Evaluation, Results and Analysis" presents the results, evaluation

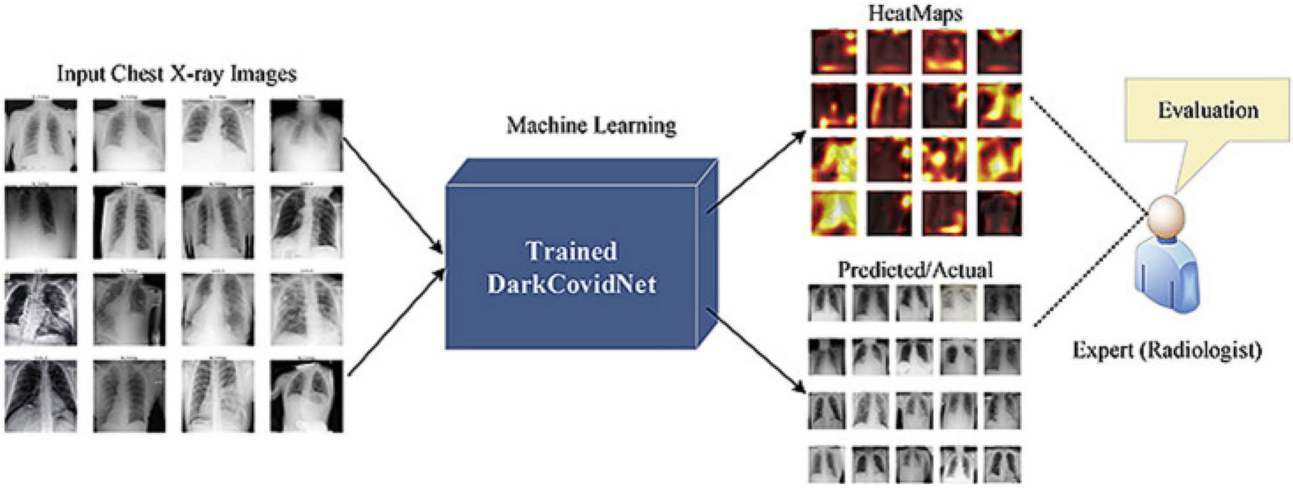


Fig. 1 Depiction of diagnosis by machine learning and a radiologist [9]

and analysis. Additionally, the best performing model is identified, and additional experiments are outlined using this model to explore the impacts of data imbalance on the model's performance. In Sect. "Discussion and Findings", the significance of the results is discussed, and a comparison of each model’s results is presented. Sect. "Research Contributions" details the research contributions of the study, Sect. "Conclusion" provides a conclusion and Sect. 8 discusses areas for future research in this area.

Literature Review

Convolutional Neural Networks

Although most characteristics of convolutional neural networks (CNNs) are comparable to those of traditional artificial neural networks (ANNs), CNNs are mostly employed in the field of pattern identification in images. To make the network better adapted for image-focused tasks while reducing the number of parameters needed to set up the model, CNNs enable encoding image-specific properties into the architecture. As can be observed in the case of complicated ANNs, CNNs also help us address the issue of overfitting.CNNs can be broken down into four layers:

1. Input layer- This layer aids the computer's ability to read images that are represented by pixel values that are kept as integers in three-dimensional arrays.
2. Convolutional layer- The central component of a CNN is the convolutional layer, which is also where most of the computation takes place. Convolution takes place at this layer. In other words, a feature detector—also known as a kernel or filter—moves across the image's receptive fields to determine whether the feature is there. Usually a 3 × 3 matrix, the feature detector is a two-dimensional array of weights. The dot product between the input pixels and the feature extractor is calculated after the filter has been applied to a portion of the image. An output array is fed the resultant number. The feature extractor navigates from pixel to pixel in the input layer and this process takes place until the whole image has been processed [7].
3. Pooling layer- This layer conducts the process of downsampling, which is essential in reducing the complexity by limiting the number of parameters in the input. The down-sampling is done by reducing the number of dimensions or the number of parameters of the input through a pooling operation in which a filter sweeps across the entire input (like the process in the convolution layer) and applies an aggregation function to the values of the input, and finally populates the output array (Fig. 2).
4. Fully connected layer- As was noted before while describing the previous layers, partially connected layers do not have a direct connection between the input image's pixel values and the output layer. In contrast, every node in the output layer of the fully connected

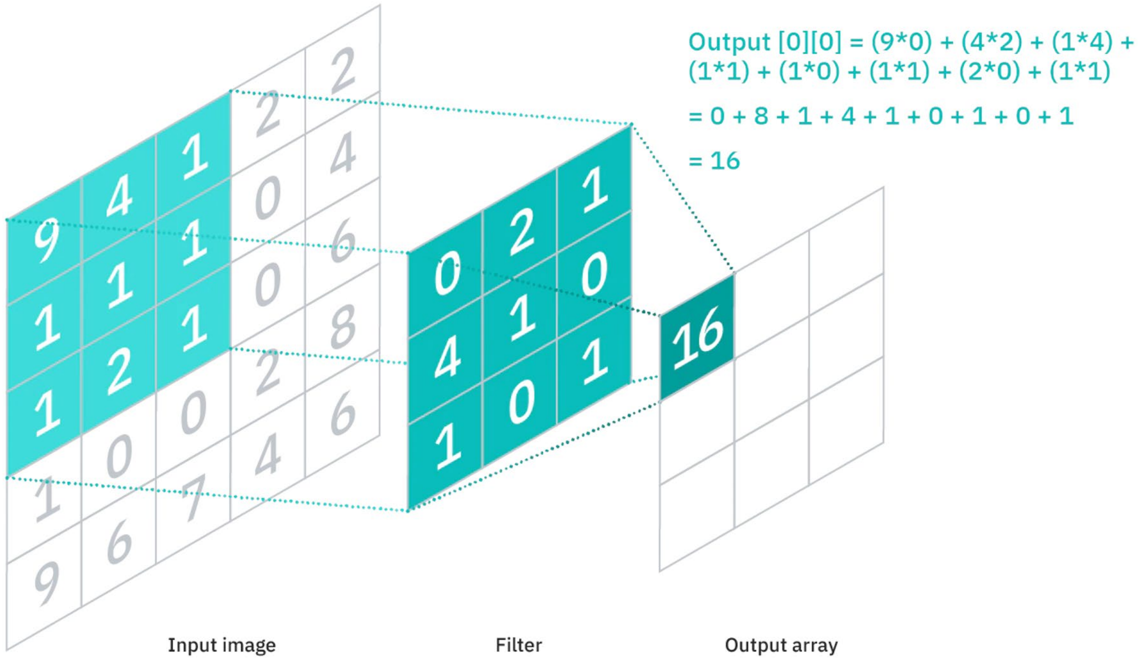


Fig. 2 The above image shows the process of convolution [7]

layer is directly connected to a node in the layer above it. Based on the features that were retrieved from the preceding layers and their various filters, this layer conducts the classification operation. [7]

CNN Data Modeling Techniques

ImageNet Large Scale, from Visual Recognition Challenge, is an image classification competition that is held every year, and it invites several data scientists and researchers from all over the world to present their models. The *ILSVRC* has set the bar for object classification and detection of hundreds of object categories from millions of images. The research paper submitted by researchers of Stanford University, University of Michigan, MIT, and UNC Chapel [13] states that the advances in object recognition have been possible due to the high number of submissions and participation seen every year. They discuss the evolution in the submission of various algorithms for the competition, such as SIFT and LBP features [23] with two non-linear coding representations and a stochastic SVM, improved Fisher vector representation Sánchez et al. [14], along with PCA dimensionality reduction [6], and data compression followed by a linear SVM among other entries winning the competition until 2011. However, there was a turning point in 2012 with the introduction of large-scale convolutional neural networks. The undisputed winner of the classification and localization tasks in 2012 was the SuperVision team. They trained a large, deep convolutional neural network with 60 million parameters on RGB values using an efficient GPU implementation and a novel hidden-unit dropout trick. Following the success of the deep learning-based method in 2012 [14], most of the entries in 2013 used deep convolutional neural networks in their submission. In 2014 and in the following years, all the top contestants relied heavily on convolutional neural networks for image classification as well as object identification and single-object localization tasks.

Developing a neural network image classification model can require significant architectural engineering. Zoph et al. [24], in Learning Transferable Architectures for Scalable Image Recognition, endeavor to build a method through which they can decide on the best model architecture based on the dataset in question. It can be very tedious and expensive to find the appropriate architecture when the dataset is large, so the researchers suggest developing the architectural building block based on a small dataset and then expanding it based on the actual larger data. Finally, they design and propose a new search space called the NASNet search space. The key feature of the NASNet search space is that it provides transferability of the learnings from the small to the large dataset. They use the small CIFAR-10 dataset to develop a CNN convolutional layer and then apply this layer to the ImageNet dataset by using multiple such layers, one above the other. All the layers used on the ImageNet dataset are not identical; rather, they have their own parameters that are used to design the final CNN. The architecture thus obtained is known as NASNet architecture. The paper also introduces a new regularization technique known as scheduled drop path that improves the ability of the NASNet models or architectures to be used across a variety of datasets. Finally, the image features learned from image classification are generically useful as the layers can identify attributes in images, such as edges, which can be transferred to other computer vision problems.

CNN Applications in the Healthcare Space

Researchers and scientists from all around the world have been working diligently to create scalable and precise technologies to identify COVID-19 and its variations. Rapid and precise COVID-19 variant detection is urgently required. Radiologists examine the picture of the lungs for COVID-19 using CT scans and X-rays. However, it necessitates radiology specialists to review specific reports, which is one of the difficulties during the epidemic. This prevents the professionals from spending time on the treatment of patients. Panwar et al. [10] propose an approach where neural networks, particularly nCOVnet**,** which comes under the class of deep-learning neural networks, can be used for detecting COVID-19 disease by analyzing the X-ray scans of patients. Using the converted rapid screening method can be used for detecting the COVID-19 virus by running the X-rays of patients through the model. It searches for visual indications found in chest radiography imaging of COVID-19 patients. According to the paper, the proposed nCOVnet model can detect a COVID-19 positive patient in under 5 s. The researchers were able to achieve a 97.62% true positive rate and assert that if the chest X-ray samples in the training data are augmented with additional samples, then the model can achieve higher accuracy while keeping the same model architecture.

In Efficient Framework for Detection of COVID-19 Omicron and Delta Variants Based on Two Intelligent Phases of CNN Models, Ghaderzadeh et al. [5] use a set of CT scan and X-ray pictures to demonstrate a general framework made up of two models created using convolutional neural networks (CNN) implementing the ideas of transfer learning and parameter optimization. The suggested model, along with all its layers, was tested using the test dataset, and the results were impressive. For the first phase, the proposed phase achieved detection sensitivity, specificity and accuracy of 0.99, 0.986, and 0.988, respectively, and for the second phase, the performance measures were 0.997, 0.9976, and 0.997, respectively.

A paper by Imaging Biomarkers and Computer-Aided Diagnosis Laboratory evaluates the performance of nine CNN configurations of CiferNet [1], AlexNet [2], and GoogLeNet [17] in randomly initialized transfer learning in high or low-resolution settings. They are applied to two specific CADe issues, namely, the classification of interstitial lung disease and the detection of thoraco-abdominal lymph nodes. They discovered that transfer learning regularly benefited their experiments. They also draw the conclusion that applications of pre-trained deep CNN pictures to CADe problems can be enhanced by investigating the performance-beneficial characteristics of hand-crafted features or by creating custom CNNs and better fine-tuning them using the target medical image dataset [15].

Ghaderzadeh et al. [4] utilized a NASNet-based algorithm in another effort to build an effective computer-aided detection system for COVID-19. The local dataset included 10,153 CT scan pictures of 190 people who were thought to have COVID-19. Following the model's fitting on the training dataset, hyper-parameter tweaking, and “topological changes to the classifier block, the suggested NASNet-based model was evaluated on the test dataset and demonstrated outstanding performance. Performance metrics for the suggested model included detection sensitivity, specificity and accuracy, which were each 0.999, 0.986, and 0.99, respectively.

Methodology

CNN Models

Since there are several models available for image classification, four popular models are trained that were found in the literature review: DenseNet201, ResNet152, VGG16, and InceptionV3 [11]. The results, findings, and comparisons across all four models were also discussed after testing.

ResNet152

ResNet152, or Residual Neural Network is a pre-trained convolutional neural network that democratized the concepts of residual learning. It has been trained on the ImageNet dataset with a depth of up to 152 layers. Like DenseNet, it has been trained on millions of images and can classify 1000+object categories. It is different from other CNNs as it relies on residual representation functions for its learning. It is better than learning signal representation, allowing the programmer to train deeper models. The formula of the residual function is defined as

$$F(x) = H(x) - x$$

where F(x) defines the residual function, x denotes the input, and H(x) is the learned parameter function. By deriving this from the standard identity mapping function H(x) = x, the residual network can converge quickly and be optimized more easily. Figure 3 shows the architecture of the ResNet block contained within the model, where the input is defined by x_{l-1}, and the output, after passing through two convolutional layers $h_l(x_{l-1})$, is computed.

ResNet152 stacks many of these blocks together to form most of the model architecture. This deep stacking approach can more effectively use the output data transmitted from the previous stacks [19]. ResNet152 then makes use of global average pooling after the convolution blocks. Each feature map is averaged spatially, resulting in a vector of fixed size, thus reducing the spatial dimensions. Finally, for image classification tasks, the fully connected layer is followed by a softmax layer as the final activation function. Given a vector z of logits for each class, the softmax function is defined as

$$softmax(z)i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{2_j}}$$

DenseNet201

DenseNet201 is a deep architecture with 201 layers. It is a pre-trained model that has been trained on the ImageNet repository using millions of images collected from a range of data sources. Similarly to ResNet, it has blocks of layers where the layers are connected to each other. When data is fed into the DenseNet CNN, the layers pass the data to each other and the dense network of connection within the layers

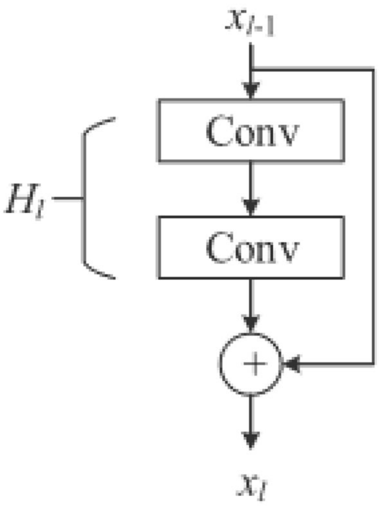


Fig. 3 ResNet block structure

in a block, and the model can learn and distinguish complex patterns. Like ResNet, each convolutional block is connected to the others using a feed-forward strategy, ensuring each layer receives direct inputs from the following layers, as defined by the following equations:

$$x_l = H_l(x_{l-1})$$

$$x_l = H_l\left(x_{l-1}\right) + x_{l-1}$$

$$x_l = H_l\left(\left[x_0, x_1, x_2, \ldots x_{l-1}\right]\right)$$

where l is the layer index, H is the non-linear operation and x_l is the feature of the lth convolutional layer.

DenseNet also implements bottleneck layers to reduce the number of input feature maps, thus reducing the computational complexity of the model. The bottleneck layers are typically made up of 1×1 blocks followed by 3×3 blocks. The growth rate hyperparameter determines how many feature maps are contributed to the following layers. A high growth rate leads to a higher number of feature maps and, therefore, increased capacity of the model. This parameter can be fine-tuned to optimize the performance of the model further. Conversely, a low growth rate leads to a lower number of feature maps and lower capacity. Finally, DenseNet makes use of global average pooling and a fully connected final softmax layer. This is done in the same way as ResNet, as explained in Sect. "ResNet152".

VGG16

VGG16, or Visual Geometry Group 16-layer [16] is another Convolutional Neural Network that was submitted to the ImageNet Large Scale Visual Recognition Challenge [13]. VGG16 consists of weighted layers: thirteen convolutional layers and three dense layers. Additionally, VGG16 also includes five max pooling layers at the end of each convolutional block layer. The first layer has 64 filters, the second layer has 128 filters, and the following layers continue to double the number of filters for each subsequent layer. The convolutional layers are followed by three fully connected layers and a softmax layer, in a similar manner to ResNet and DenseNet-derived models. Figure 4 provides a visual representation of the VGG16 architecture.

Unlike traditional neural networks, VGG16 forgoes matrix multiplication to transition between convolutional layers. Instead of using 2-dimensional weight matrices, VGG16 uses 3-dimensional filter tensors with the addition of a weighted bias and ReLU activation functions, which is defined as

$$f(x) = \max(0, x)$$

where x is the neuron input to the function, the ReLU function outputs the input value if $x \geq 0$; otherwise, the function returns 0. This can be expressed mathematically as

$$f(x) = \begin{cases} x, if\ x \geq 0 \\ 0, otherwise \end{cases}$$

InceptionV3

InceptionV3 is a convolutional neural network model that is 22 layers deep. It has been trained using millions of images in ImageNet data and can categorize more than 1000 objects. It is available with ImageNet weights. Initially, InceptionV3 started as a module for its precursor, GoogLeNet and is based on the seminal work by Szegedy et al. [18]. Similarly to VGG16, it was first introduced as part of the ImageNet Large Scale Visual Recognition Challenge, where it demonstrated a 78.1% accuracy on the ImageNet dataset.

Like other CNN models, InceptionV3 mostly uses convolutional layers for feature extraction. The convolution operation in InceptionV3 works by sliding kernels over the feature map and computing the dot product between the kernel and the overlapping input region. More specifically, this can be defined as

$$(f * w)(i, j) = \sum_{m,n} f(m, n) \bullet w(i - m, j - n)$$

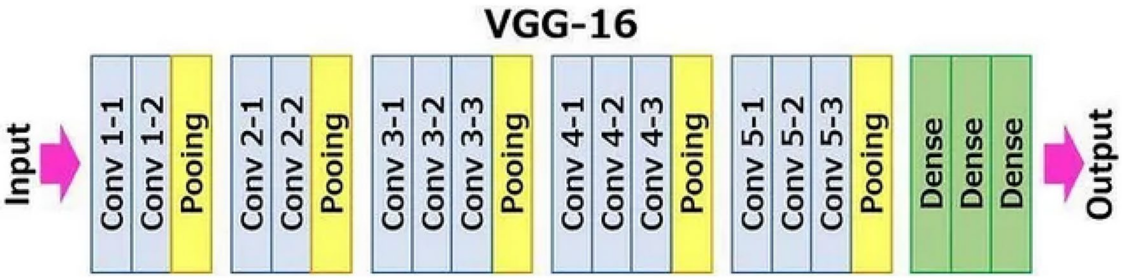


Fig. 4 VGG model architecture

where f is the input map, w is the convolutional kernel and $(f * w)(i,j)$ is the output value at the position (i,j) in the feature map.

The convolutional layers are made up of inception modules, built up of parallel convolution operations with kernel sizes of 1 × 1, 3 × 3 and 5 × 5. Additionally, optional 1 × 1 kernels can be included if dimensionality reduction is required. Similarly to the other models discussed in Sect. "CNN Models", InceptionV3 also utilizes global average pooling and the softmax activation function in the fully connected layers.

Data Collection

The data required for this study was to collect CT scan images of patients classified as having positive or negative diagnoses. The dataset is split into training, validation and testing groups in the ratio of 20:2:3 where the training and validation datasets will go into the training of the model, while the testing dataset will be used to evaluate the model using a standardized set of performance metrics.

Once the idea is conceptualized, the next step is to obtain a data source for CT scan reports along with patient outcomes as positive or negative. The CT scans used in this research are sourced from a dataset uploaded from a hospital in Iran for a paper [3]. This is a large public COVID-19 (Omicron and Delta Variant) lung CT scan dataset. According to Eshraghi [3], "It contains 14,482 CT scans which include 12,231 positive cases (COVID-19 infection) and 2251 negative ones (normal and non-COVID-19). Data is available as 512 × 512px JPG images and has been collected from real patients in radiology centers of teaching hospitals in Tehran, Iran. The aim of this dataset is to encourage the research and development of effective and innovative methods such as deep CNNs, which can identify if COVID-19 infects a person through the analysis of his/her CT scans."

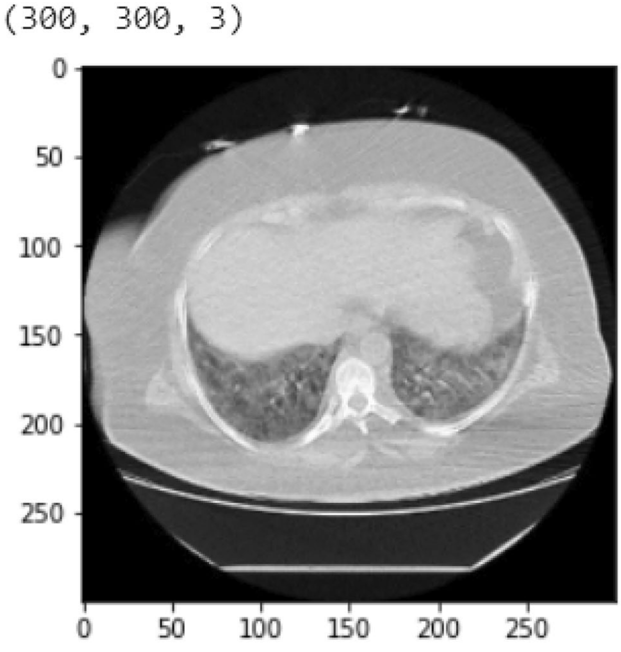
The data is imbalanced and appropriate measures need to be taken to balance the data before training the model in the event of low values of performance measures of the models. The dataset has 12,233 COVID-19 images and 2251 images of non-COVID patients. The data is imbalanced as the number of non-COVID images is far less than the number of COVID images. Therefore, oversampling techniques such as SMOTE or ADASYN need to be implemented to address this shortcoming.

Data Preprocessing

It is important to resize all the images before using them for model training so that equal memory is allocated while training or processing the images. This helps maintain consistency and saves time from memory reallocation. The images are resized to a size of 300 × 300 and all images are converted to grayscale. Figure 5 shows an example of one of the images.

The images in the dataset are divided into several groups: (a) train, (b) validation and (c) test sets. Out of 14,484 images, 80% were used for training, while 8% and 12% of the images were used for validation and testing, respectively.

The images that are present in the train, validation, and test sets were transformed using the Keras image data generator. Rescale [0,1], vertical flip, horizontal flip, and random rotations with a rotation range of 10 degrees were used to augment the images in the training set. At the same time, only rescale was used for the validation and the testing images because they were not used for the learning of the model and should be representative of new diagnostic images. An image data generator is used to transform the images in the training, validation, and testing datasets on a random basis and store the newly transformed images in the train generator, valid generator, and test generator, respectively. The batch size is set to 32 in all three generators. The augmented images are used in the training and validation set to train and validate the model simultaneously. Figure 6 shows random images from the training generator. The first row of images shows the COVID-19 images in the trained generator, while the second row of images shows the non-COVID images present in the training data.



```
(300, 300, 3)
```

Fig. 5 A COVID positive patient's CT scan image after balancing

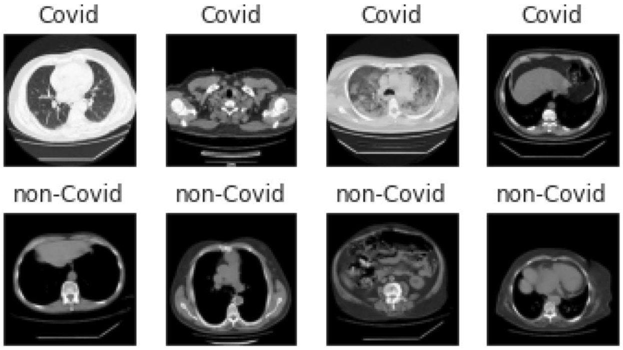


Fig. 6 COVID and non-COVID images after applying the Image fata generator on the training data

Model Tuning

This is the most important step in the entire process as the architecture of the convolution neural network is defined in this stage. The pre-trained CNN is loaded from Keras with ImageNet weights. In the first iteration, DenseNet201 is loaded with ImageNet weights as the pre-trained base. The CNN model learns the parameters using a sizable dataset to represent the global and local information in the image. Each model architecture includes a different kind of layer and activation function to better depict characteristics than features created by humans (Fig. 7).

A pre-trained base model for image processing DenseNet201, which is 201 layers deep, is used for the model. DenseNet201 is a pre-trained CNN that is trained using more than a million images from the ImageNet database. The ImageNet database is a huge repository of 14 million hand-annotated images indicating the objects pictured. More than one million images in ImageNet also have bounding boxes. ImageNet is widely used for training models in the field of visual object recognition software research. It has been trained to classify images into 1000 + object categories, and the initial and middle layers can be reused, with the final layers needing to be added.

A layer of 2-dimensional global average pooling is added to reduce the spatial dimension of the tensor by averaging all the values across the entire matrix for each input channel. In addition, the following layers are added to the pre-trained DenseNet201:

1. Dense layer with reLu activation
2. Batch Normalization layer
3. Dropout layer
4. Dense layer with softmax activation

Model Training

The architecture defined above is trained using the train and validation generator. The following hyper-parameters are set for the initial model training:

1. Number of epochs: 40
2. Steps per epoch: 275
3. Verbose: 2
4. Early callbacks set to avoid overfitting

The batch size is 32 for the trained generator, which is used for training the DenseNet201 model. The train generator has 11,587 images. Hence, there will be 362 batches of the training data. The validation data has 1159 images with a batch size of 32, which means the step size for the validation generator is 37 batches. The accuracy of the model is tested on the training and the validation data after each epoch and the training proceeds to the next epoch. This continues until the accuracy of the validation data gets saturated and does not show any improvement. In this iteration of DenseNet201, the model runs for 13 epochs and the accuracy on the training and validation set is 99.74% and 98.62%, respectively.

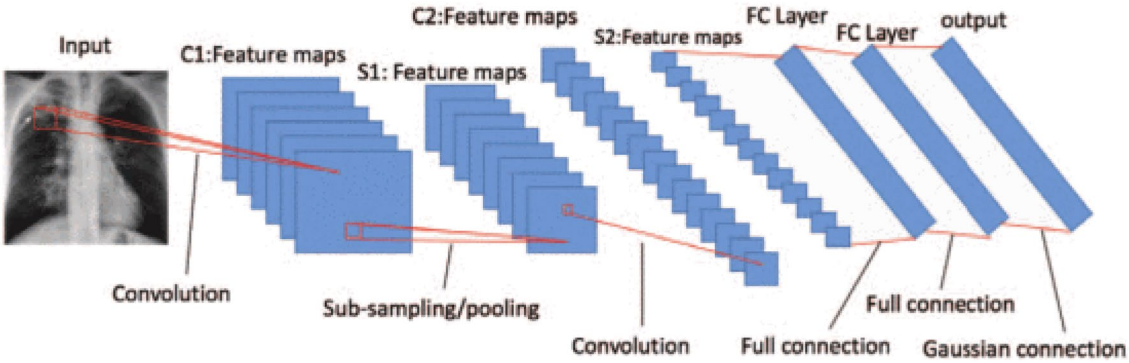


Fig. 7 CNN architecture (A LeNet model) for TB classification [8]

Evaluation, Results and Analysiss

The model was tested on the test generator, which has 12% of the images, i.e., 1738 images. There are 1486 COVID-19 and 252 non-COVID images in the test generator. The performance measures that were used to evaluate model performance are F1 score, accuracy, precision (positive predictive value), recall (sensitivity, hit rate, true positive rate), specificity (true negative rate), negative predictive value, Fall out (false positive rate), and false negative rate.

The F1 score is a metric that is calculated by taking the harmonic mean of recall and precision, where the recall is the number of true positives divided by the total actual positive cases, and the precision is the number of true positives divided by the total predicted positive cases. An F1 score can range from 0 to 1, where a score closer to 1 indicates better performance and a score closer to 0 indicates worse performance.

The accuracy of the model on all the 1738 images in the test data was found to be 98.73%. The training and validation loss was displayed for the model while training, as shown in Fig. 8. Both the graphs seem to converge and become parallel after 8 epochs.

The plot for training and validation accuracy is also shown in the figure below. As mentioned previously, the training of DenseNet201 terminates after 13 epochs when the accuracy of the training and validation set reaches 99.74% and 98.62%, respectively, and the same can be verified from Fig. 9.

Subsequently, the confusion matrix of the test images is computed and shown below in Fig. 10. It can be seen clearly that 1485 COVID-19 images in the test set have been correctly labeled/predicted as COVID-19, while only one image has been incorrectly predicted as non-COVID. From the non-COVID set, 231 out of 252 images have been correctly predicted as non-COVID, while 21 get mislabeled as positive COVID-19 infections.

The normalized confusion matrix is shown in Fig. 11 and the percentages of correct vs. incorrect prediction in each of the COVID and non-COVID buckets can be seen.

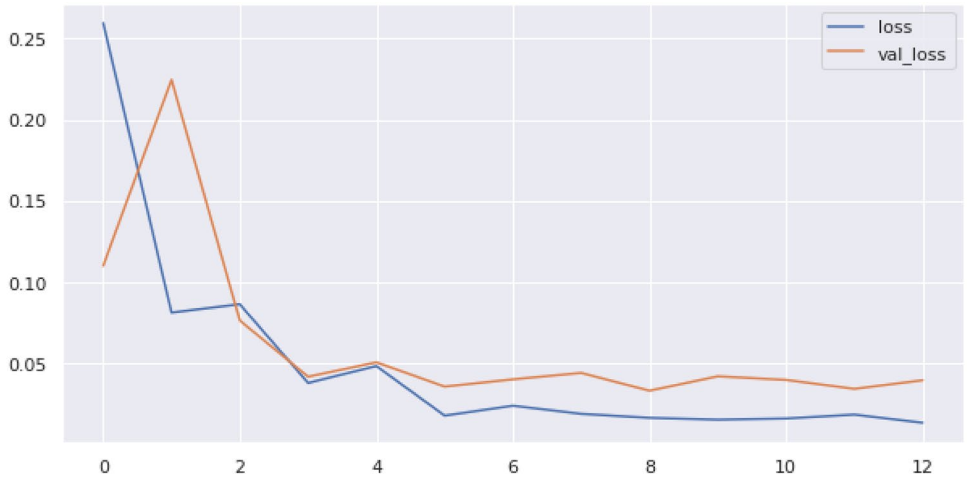


Fig. 8 Plot of training vs. validation loss with the number of epochs for DenseNet201

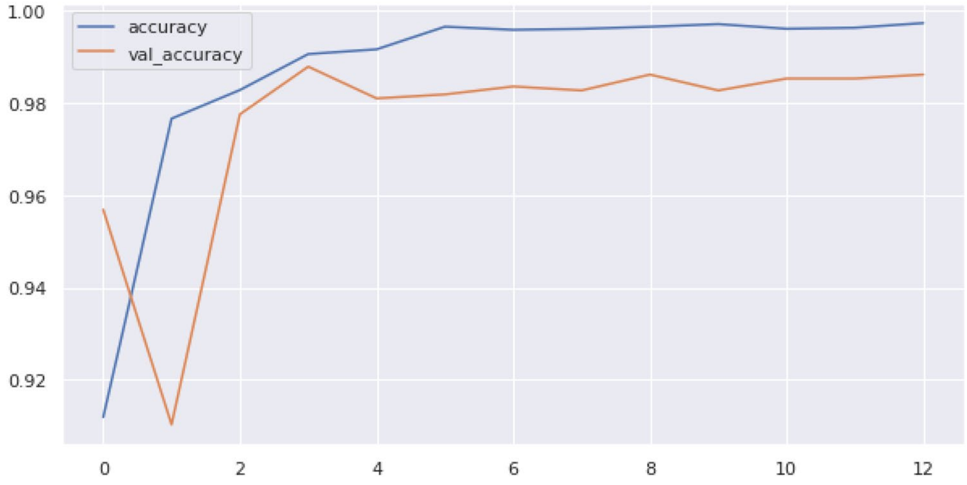


Fig. 9 Plot of training vs. validation accuracy with the number of epochs for DenseNet201

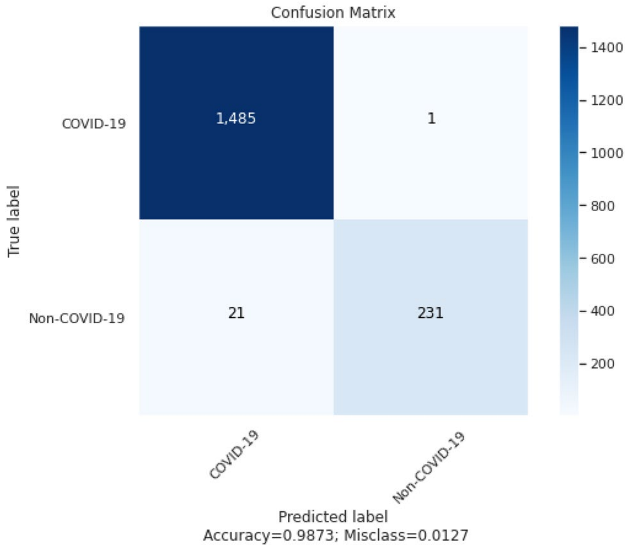


Fig. 10 Confusion matrix obtained from test data using DenseNet201

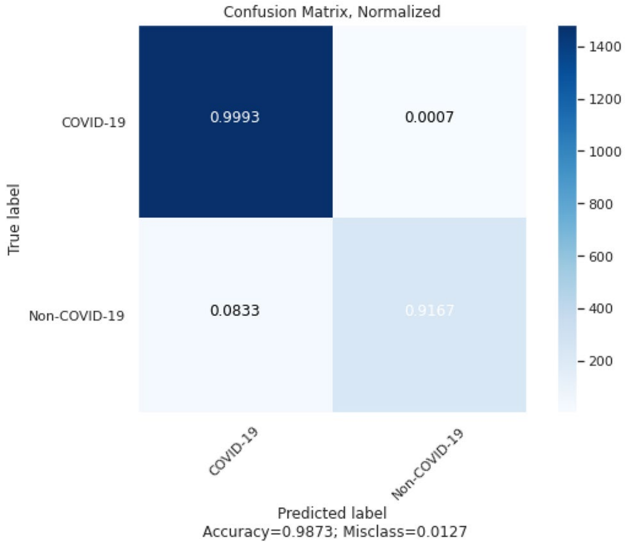


Fig. 11 Normalizing the confusion matrix obtained from test data using DenseNet201

a. 99.93% of the COVID images are correctly predicted by the model.
b. 91.67% of the non-COVID images are correctly predicted by the model.

The performance of the model seems to be better when COVID-19 images are put up for prediction. This could be because non-COVID images are the minority class in the dataset, i.e., in the training set, there were 11,587 images, out of which 9774 images were COVID images, while only 1813 images were non-COVID. This led the model to be trained better for the COVID class of images than non-COVID.

The class weight method to handle imbalanced datasets is used to resolve this issue. This will give more weight to the minority class of images (which are non-COVID images) while training the model, providing a more balanced dataset.

Class-weight for Addressing Data Imbalance

The initial bias of COVID to non-COVID images is computed by dividing the number of COVID images in training data by the number of non-COVID images in training data and is calculated to be 1.69. Additionally, the weight of each class of images is also computed: the weight for class 0 (non-COVID) is a lot higher (3.22 for non-COVID) than the weight for class 1 (0.59 for COVID). This is necessary because there are fewer non-COVID images, so each non-COVID image will be weighted more to balance the data, as CNN models work best when the training data is balanced.

The model has been trained once again with the same hyper-parameters as before, but this time, the class weight argument was added within the model fitting function while training the model. The model is trained for 11 epochs before the training terminates.

After the training is complete, the accuracy of the model is tested on the test set. The model shows an accuracy of 97.24% on the 1,738 images in the test generator.

Next, the training and validation loss is plotted in Fig. 12, which converges after 6 epochs.

Also, the training and validation accuracy of the model seems to be parallel and saturated after 6 epochs (Fig. 13).

Furthermore, the confusion matrix is obtained on all 32 batches or 1,738 images of the test generator.

a. It was found that 1,465 COVID-19 images in the test set have been correctly predicted as COVID-19, while no image has been predicted as non-COVID.
b. From the non-COVID set, 225 out of 273 images have been correctly predicted as non-COVID, while 48 get mislabeled as COVID-19 (Fig. 14).

The confusion matrix is normalized to scale the number of images in each true label, i.e., COVID and non-COVID, between 0 and 1.

a. 100% of the COVID images are correctly predicted by the model.
b. 82.42% of the non-COVID images are correctly predicted by the model.

All the performance measures were computed from the confusion matrices above and shown in the table below (Table 1). COVID-19 in the True label is considered a

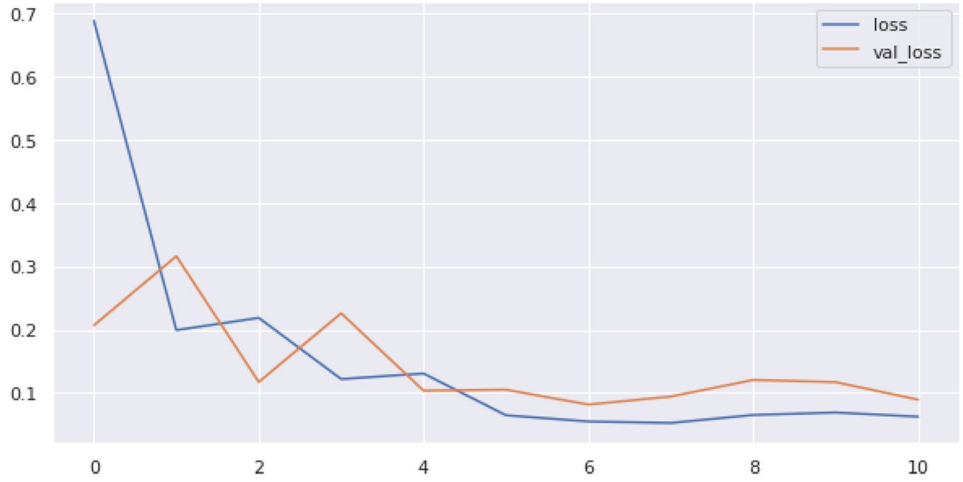


Fig. 12 Plot of training vs. validation loss with a number of epochs for DenseNet201 after balancing the minority class

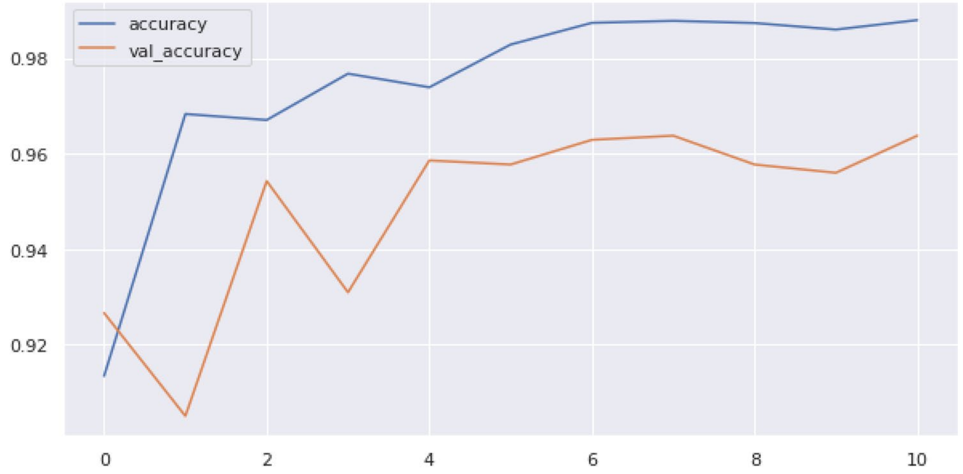


Fig. 13 Plot of training vs. validation accuracy with a number of epochs for DenseNet201 after balancing the minority class

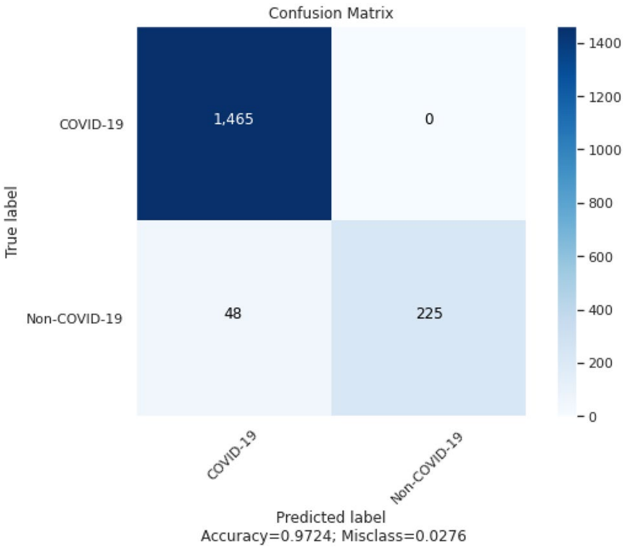


Fig. 14 Confusion matrix obtained from test data using DenseNet201 after balancing of minority class

positive outcome, while non-COVID-19 is a negative outcome (Fig. 15).

Discussion and Findings

The performance of the model is observed when trained under two different conditions (with and without the dataset balancing technique), and it is shown that DenseNet201 works well with imbalanced data. The F1-score reduced to 0.94 from 0.97, Accuracy reduced to 97.23% from 98.73%, Recall rose to 100% from 99.93%, and Specificity stood at 82.42%, down from 91.67% (Fig. 16).

The model's performance did not change massively except for some noticeable changes in predicting negative outcomes. However, this change is ostensibly counter-intuitive because the prediction of negative outcomes turned out worse after using class weight to balance the data. Ideally, the *true negative rate* should have improved when class

Table 1 Performance measures for DenseNet201 before and after balancing the image classes

Performance measures Balanced vs. Imbalanced dataset	DenseNet201 Imbalanced image classes Training for 13 epochs	DenseNet201 Used class weight to handle imbalance of image classes Training for 11 epochs
F1-score (macro avg)	0.97	0.94
Accuracy	98.73	97.23
Precision (positive predictive value)	98.60	96.83
Recall (sensitivity, hit rate, true positive rate)	99.93	100
Specificity (true negative rate)	91.67	82.42
Negative predictive value	99.57	100
Fall out (false positive rate)	0.07	0
False negative rate	8.33	17.58

weight was used to give more weight to non-COVID images while training. The F1-score is indicative of the performance of the model in both cases, and it can be ascertained that DenseNet201 performed well with imbalanced data with an F1-score of 0.97 as against an f1-score of 0.94 when fed with balanced data.

It should be noted that the model was trained for 13 epochs previously. In contrast, when class weight was used, it trained for 11 epochs, which could have led to a subtle decline in most of the model performance measures and a remarkable decline in predicting non-COVID images accurately. At this point, it can be concluded that DenseNet201 works well with imbalanced data, and one should focus on reaching a decent number of epochs while training the model. This can be achieved by increasing the size of the validation data generator while still having a large training data generator.

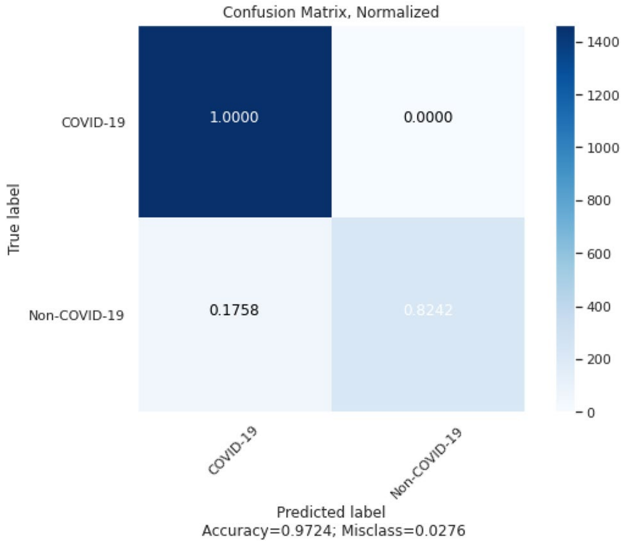


Fig. 15 Normalizing the confusion matrix obtained from test data using DenseNet201 after balancing the minority class

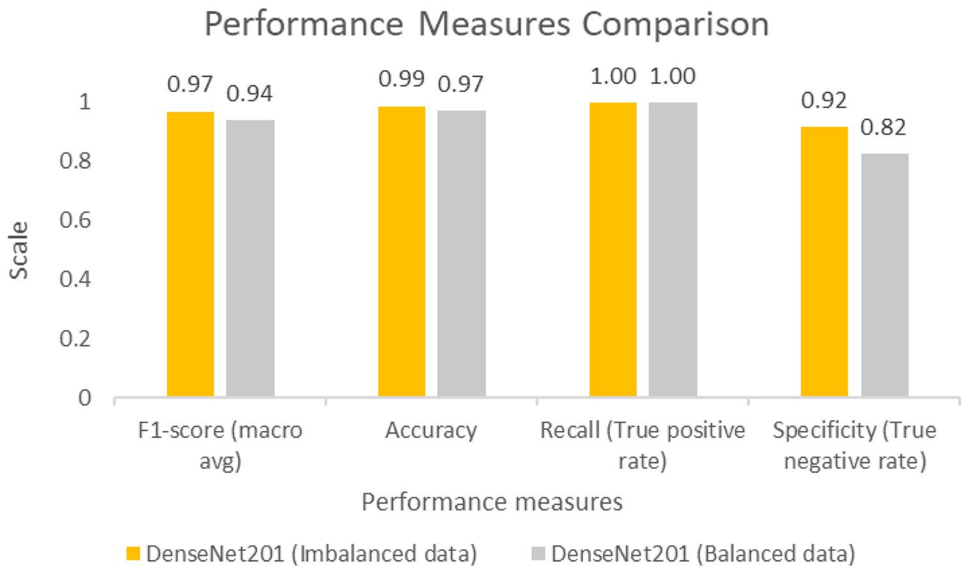


Fig. 16 Comparison of F1-score, accuracy, recall, and specificity for DenseNet201 before and after balancing the minority class

Performance of Other CNN Models

There are other CNNs that researchers have developed and show remarkable results in applications of image classification. Three CNN models were explored, and layers were added on top in the same manner as with DenseNet201. The evaluation metrics obtained from the models were recorded in the table below and a detailed discussion of their performance was carried out. Preliminary experiments were conducted in each instance using different numbers of epochs to determine the optimal number of epochs for each model. The results presented in Table 2 show only the best results using the determined optimal number of epochs in accordance with the preliminary experiments. If the number of epochs is too small, the model can suffer from underfitting as a result. On the other hand, if the number of epochs is too high, the model is at risk of overfitting the training data.

The training, validation, and test data had the same number of images across all the models, as the split ratio was kept consistent across all the models. It can be clearly seen that the accuracy is well above 90% in all the models explored in the study (Fig. 17).

Model with Best Performance

DenseNet201, ResNet152 and InceptionV3 models give accuracy above 97%. Although this might seem to have a very high accuracy, other measures also need to be examined, which focus on the measures by outcomes, i.e., COVID-19 (positive) and non-COVID (negative) outcomes. This is important since the model is required to perform well for all patients, irrespective of whether they have COVID-19 or not. Recall is important in this aspect as COVID-19 patients being identified as COVID-19 positive with high accuracy is crucial to model performance. The healthcare system cannot really afford to miss out on true positive patients being tagged non-COVID by the model. At the same time, it is also essential to maintain high enough accuracy of true negative or non-COVID patients being predicted as COVID-negative. The F1-score is highest in the case of both DenseNet201 (0.97) and InceptionV3 (0.97), which indicates that the performances of these models are the best in this regard (Figs. 18 and 19).

InceptionV3 does the best in terms of overall accuracy (98.38%), precision (99.72%), recall (98.36%), and

Table 2 Performance measure for the four CNNs without balancing image classes

Performance measures Imbalanced dataset	DenseNet201 Trained for 13 epochs	ResNet152 Trained for 10 epochs	VGG16 Trained for 8 epochs	InceptionV3 Trained for 21 epochs
F1-score (macro avg)	0.97	0.95	0.81	0.97
Accuracy	98.73	97.23	91.25	98.38
Precision (positive predictive value)	98.60	98.34	90.59	99.72
Recall (Sensitivity, hit rate, true positive rate)	99.93	98.34	99.79	98.36
Specificity (true negative rate)	91.67	91.69	50.17	98.54
Negative predictive value	99.57	91.69	98.04	91.83
Fall out (false positive rate)	0.07	1.65	0.20	1.64
False negative rate	8.33	8.30	49.83	1.46

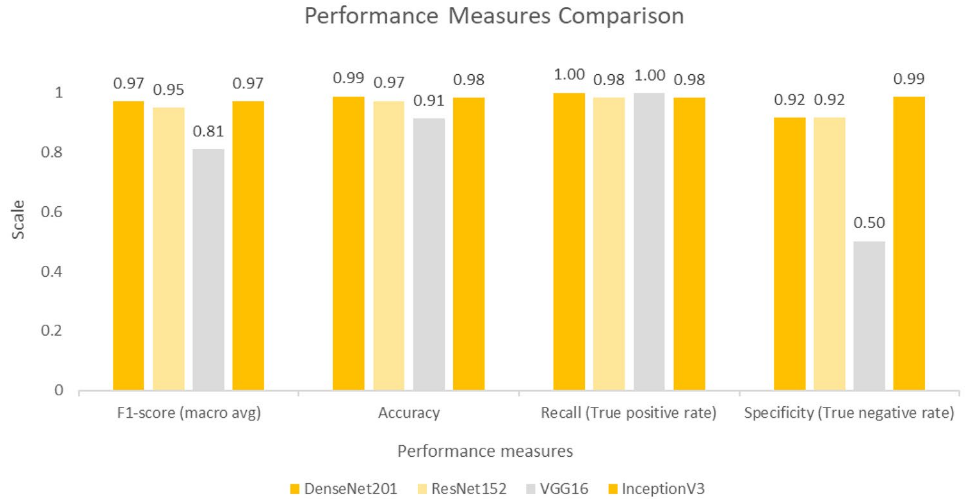


Fig. 17 Comparison of F1-score, accuracy, recall, and specificity for the 4 CNN models- DenseNet201, ResNet152, VGG16, and InceptionV3

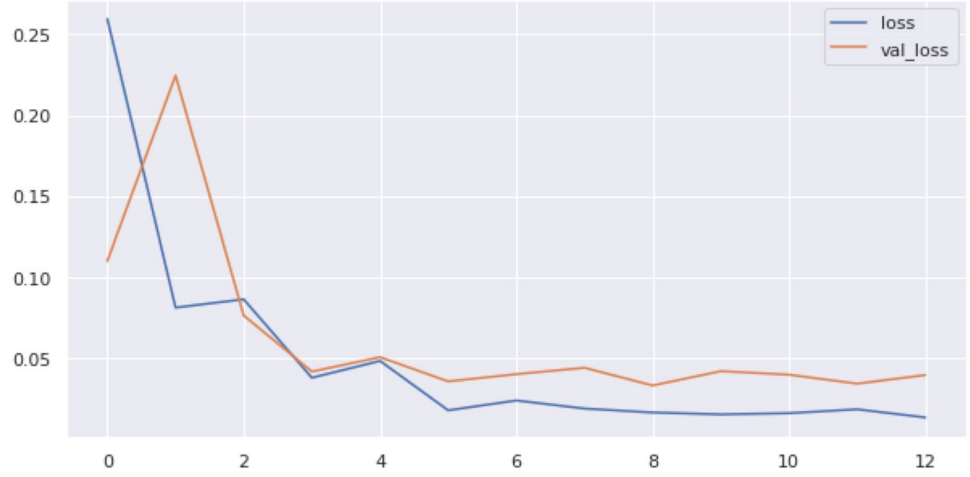


Fig. 18 Plot of train vs. validation loss of DenseNet201 (best-performing model)

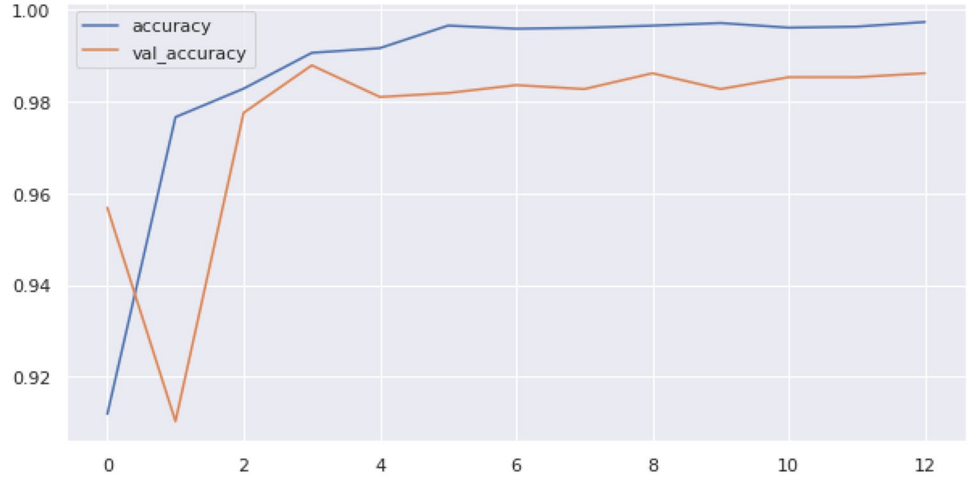


Fig. 19 Plot of train vs. validation accuracy of DenseNet201 (best-performing model)

specificity (98.54%). However, InceptionV3 has also run for 21 epochs, which no other model had reached in training during this research to make it standardized across all models to get a fair comparison. DenseNet201 is the model with the second highest number of epochs while training, i.e., 13 epochs, and shows the high value of overall accuracy (98.73%), precision (98.60%), recall (99.93%), and specificity (91.67%).

Comparisons with Existing Work

When compared with existing studies, the work conducted in this study performs well. From the conducted experiments in this study, DenseNet201 and Inception V3 performed similarly well. Therefore, these two models are used as a basis for comparison. To get the fairest comparison, the chosen models were only compared with other studies focused on the implementation of CNN models for COVID-19 diagnosis that used lung scans as their data source. Most of the work detailed in Table 3 was highlighted by Ghaderzadeh et al. [4] unless specified otherwise. It should be noted that the studies used for comparison mostly used a different image dataset than the one used in this study. Therefore, a direct comparison between

Table 3 Comparison of results between DenseNet201, InceptionV3 and other studies

	F1-score	Accuracy	Precision	Recall
DenseNet201	97.0	98.8	98.6	99.9
InceptionV3	97.0	98.4	99.7	98.3
SqueezeNet	95.2	95.1	94.2	96.2
ShuffleNet	97.5	97.5	96.1	99.0
GoogLeNet	91.8	91.7	90.2	93.5
AlexNet	93.6	93.7	94.9	92.2
ResNet50	95.0	94.9	93.0	97.1
Xception	98.8	98.8	99.0	98.6
AdaBoost	95.1	93.6	96.7	95.1
Decision Tree	79.8	79.4	76.8	83.1
COVID CT-Net [22]	90.0	90.7	88.5	85.0
NASNet [4]	99.5	99.4	99.6	99.8

these can not be perfect. Nevertheless, it shows that the DenseNet201 and InceptionV3 models used in this study compare favorably to most other models. Furthermore, it may be possible to optimize these two models further through better hyperparameter optimization.

Research Contributions

This research aims to discuss the structure of today's healthcare and the role of AI and Machine Learning in improving the quality and cost efficiency of the system. There can be several areas in which ML techniques can be incorporated. However, this paper focuses on diagnosing patients for a viral infection using CT scan images. Around 400 research papers and web articles were perused during this research, out of which 40 were selected for literature review.

It was found that healthcare around the world needs to adopt a Primary Health Care (PHC) approach to deliver essential health services to every human being. Moreover, it is necessary to set up facilities like ambulatory surgery centers (ASCs) for specific surgeries or procedures that are expensive in hospitals to better manage healthcare costs by catering to procedures at a larger scale while maintaining the quality and cutting unnecessary inpatient expenses.

ML methods can be used in healthcare by insurance companies for claims data processing, by pharma companies in the clinical trial phase to test the efficacy of drugs, and by providers to understand the clinical progression of patients in a disease area and identify key opinion leaders in the market. The application of ML does not end here, but these are only a few examples of where ML is implemented today. It can be used more proactively in diagnosing patients, and keeping this in mind, the performance of 4 different image identification models was tested with pre-trained ImageNet weights. The models perform well, and the learnings from this research can be used by facilities at a larger scale to tackle the problem of diagnosing patients.

This research can be used as a reference by Parent Health Systems to understand the direction in which they should move to adapt to the future demands of a growing old-age population, especially in terms of digitizing systems and setting up facilities like ASCs. This research also serves as valuable work and proof of using ML models for processing diagnostic images for medical technology providers. They might choose to use the best-performing model or a combination of top-performing models to declare results with higher confidence and intervene for patients where the result from multiple models is inconclusive.

Conclusion

Summary of Contributions

The Keras library has models available with pre-trained ImageNet weights for DenseNet201, DenseNet121, DenseNet169, InceptionV3, InceptionResNetV2, ResNet152, ResNet152V2, ResNet101, ResNet50, Xception, VGG16, VGG19, MobileNet, MobileNetV2, and others. From the literature review conducted for this research, ResNet152, VGG16, and InceptionV3 were the most popular models among researchers and were chosen along with DenseNet201 to evaluate their performance on the image classification dataset. It was found that DenseNet201 worked remarkably well with imbalanced images; even when image balancing techniques such as class weight were used, the model did not show any improvement in performance results. The efficiency and performance of other model architectures present within the Keras library were also evaluated, showing competitive results when compared with existing studies. DenseNet201 and InceptionV3 performed especially well, achieving accuracy scores of 98.8% and 98.4%, precision scores of 98.6% and 99.5%, and recall scores of 99.9% and 98.3%, respectively.

Future Work

The four models used in this research are DenseNet201, ResNet152, VGG16, and InceptionV3, and most of these demonstrated remarkable results. However, there is scope to expand this research and take its findings to the next level.

The split of the train, valid, and test data should be altered, and multiple iterations should be run. As discussed earlier, the performance of the models was seen to be dependent on the number of epochs. The epochs are dependent on the training and validation data used to train the model, and adding more images to the training and validation set might increase the model training time and epochs. Another way to achieve the same is by changing the patience number in the callbacks rule for early stopping.

Taking a multi-modal approach may also prove more effective than focusing solely on CT scan images. Combining information from other imaging modalities, such as MRI scans and X-rays, could potentially provide a more comprehensive understanding of COVID-19 diagnosis. Training CNN models that can effectively fuse multiple image types could improve the effectiveness of the models, leading to a more accurate diagnosis.

There is still a dependency on physicians to diagnose patients despite positive outcomes and confidence from research, mainly because the healthcare infrastructure has been under expectations towards digitalization, and patients prefer to physically visit facilities and get the personal touch of being examined by a doctor. The black-box nature of neural networks makes it challenging for doctors to understand how these models operate. Designing interpretable CNN models can help provide humanlike insights in this regard, improving transparency and interpretability. For example, attention mechanisms can be implemented into model architecture, allowing for the potential to highlight regions of importance in medical scans. This would be particularly useful at providing healthcare professionals with a deeper understanding of how the models operate, as it is unrealistic to expect physicians to be experts in machine learning techniques.

Novelty and Adoption

The research conducted in this study provides a comprehensive evaluation and comparison of multiple CNN models for COVID-19 diagnosis. This provides a useful reference or starting point for healthcare professionals when choosing which model(s) to implement to aid their diagnosis. This approach is targeted at countries or regions with fewer resources, so that medical researchers can use them more easily and effectively, providing an alternative but trusted, convenient, and easy-to-use method for analysis.

The adoption of CNN models has previously facilitated significant technological advancements in medical imaging analysis. Researchers have developed innovative algorithms and architectures tailored for COVID-19 detection, contributing to the state-of-the-art medical image analysis and AI applications in healthcare by streamlining the diagnostic process.

The adoption of CNN models in COVID-19 detection reflects the interdisciplinary nature of research in this field, involving expertise from computer science, medical imaging, epidemiology, and clinical medicine. This interdisciplinary approach has enriched the research landscape, fostering synergies between different disciplines and driving innovation at the intersection of technology and healthcare.

Author Contributions Victor Chang: Writing – review & editing, Writing – original draft, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. Siddharth Mcwann: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis. Karl Hall: Writing – review & editing, Supervision, Conceptualization, Validation. Qianwen Ariel Xu: Writing – review & editing, Validation. Meghana Ashok Ganatra: Writing – review & editing, Software.

Funding This work is partly supported by VC Research (VCR 0000174) for Prof Chang.

Data Availability Statement Authors have no data ownership or access to data anymore.

Declarations

Conflict of Interest Not applicable.

Informed Consent Not applicable.

Research Involving Human and /or Animals Not applicable.

References

1. Ai T, Yang Z, Hou H, Zhan C, Chen C, Lv W, Tao Q, Sun Z, Xia L. Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. Radiol Soc Public Health Emerg Collect. 2020;296(2):1–9 https://doi.org/10.1148/radiol.2020200642
2. Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Van Esesn BC, Awwal AA, Asari VK. The history began from alexnet: A comprehensive survey on deep learning approaches. arXiv preprint arXiv:1803.01164. 2018
3. Eshraghi MA. COVID-19 Omicron and delta variant lung CT scans. Available at: https://www.kaggle.com/datasets/mohammadamireshraghi/COVID19-omicron-and-delta-variant-ct-scan-dataset. 2021)
4. Ghaderzadeh M, Asadi F, Jafari R, Bashash D, Abolghasemi H, Aria M. Deep convolutional neural network-based computer-aided detection system for COVID-19 using multiple lung scans: design and implementation study. J Med Internet Res. 2021;23(4):1–32 https://doi.org/10.2196/27468"
5. Ghaderzadeh M, Eshraghi MA, Asadi F, Hosseini A, Jafari R, Bashash D, Abolghasemi H. Efficient framework for detection of COVID-19 omicron and delta variants based on two intelligent phases of CNN models. Comput Math Method Med. 2022. https://doi.org/10.1155/2022/4838009
6. Hasan BM, Abdulazeez AM. A review of principal component analysis algorithm for dimensionality reduction. J Soft Comput Data Min. 2021;2(1):20–30
7. IBM Convolutional Neural Networks. Available at: https://www.ibm.com/topics/convolutional-neural-networks. 2022
8. Kannan KVDMJ. A method to apply Convolution Neural Network model to detect and classify Tuberculosis (TB) manifestation in X-ray images. IJRAR-Int J Res Anal Rev (IJRAR). 2021;8(1):125–136

9. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput Biol Med. 2020;121:1-11. https://doi.org/10.1016/j.compbiomed.2020.103792
10. Panwar H, Gupta PK, Siddiqui MK, Morales-Menendez R, Singh V. Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet Chaos. Solitons Fract. 2020;38:1–8. https://doi.org/10.1016/j.chaos.2020.109944
11. Li Z, Liu F, Yang W, Peng S, Zhou J. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans Neural Networks Learn Syst. 2021;33(12):6999–7019.
12. Rong G, Zheng Y, Chen Y, Zhang Y, Zhu P, Sawan M. COVID-19 diagnostic methods and detection techniques: a review. Ref Modul Biomed Sci. 2021;3:17–32. https://doi.org/10.1016/B978-0-12-822548-6.00080-7
13. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC. Imagenet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–252. https://doi.org/10.1007/s11263-015-0816-y
14. Sánchez J, Perronnin F, Mensink T, Verbeek J. Image classification with the fisher vector: theory and practice. Int J Comput Vision. 2013;105:222–245
15. Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans Med Imaging. 2016;35(5):1285–1298 https://doi.org/10.1109/TMI.2016.2528162
16. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. 2014
17. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 1–9)
18. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 2818–2826).
19. Tang Y, Gu L, Wang L. Deep stacking network for intrusion detection. Sens. 2021;22(1):25
20. Wang B, Jin S, Yan Q, Xu H, Luo C, Wei L, Zhao W, Hou X, Ma W, Xu Z, Zheng Z. AI-assisted CT imaging analysis for COVID-19 screening: building and deploying a medical AI system. Appl Soft Comput. 2021;98:1–11.https://doi.org/10.1016/j.asoc.2020.106897
21. World Health Organization. Global health expenditure database. Available at: https://apps.who.int/nha/database. 2022.
22. Yazdani S, Minaee S, Kafieh R, Saeedizadeh N, Sonka M. Covid ct-net: Predicting covid-19 from chest ct images using attentional convolutional network. arXiv preprint arXiv:2009.05096. 2020.
23. Yuan X, Yu J, Qin Z, Wan T. A SIFT-LBP image retrieval model based on bag of features. In IEEE international conference on image processing 2011 (pp. 1061–1064)
24. Zoph B, Vasudevan V, Shlens J, Le QV. 'Learning transferable architectures for scalable image recognition', 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, (pp. 8697–8710). 2018. Available at: https://doi.org/10.1109/CVPR.2018.00907